DOCUMENT RESUME

ED 254 199 IR 011 511

AUTHOR Dale, Evelyn J.
TITLE Logo's Problem-Solving Potential.
PUB DATE [83]
NOTE 13p.; Paper presented at the National Educational Computing Conference (Dayton, OH, June 13-15, 1984).
PUB TYPE Viewpoints (120) -- Speeches/Conference Papers (150)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS *Computer Assisted Instruction; Computers; *Educational Objectives; *Intellectual Development; Intelligence Differences; *Problem Solving; *Programing; Programing Languages; *Teaching Methods
IDENTIFIERS Blooms Taxonomy; Computer Uses in Education; Feuerstein (Reuven); *LOGO Programing Language

ABSTRACT

Given the uncertainty of the future and the rapidity with which computer technology is changing, a generalist position on the objectives of educational computing is desirable. This position insists that learning how to think and solve problems is the foundation of education and suggests that basic learning needs to be an integral part of the broader educational purpose. Logo computer language promotes learning by being accessible, interactive, and extensible. Children use Bloom's six levels of thinking (knowledge, comprehension, application, analysis, synthesis, and evaluation) when working with Logo. Reuven Feuerstein, whose work assumes that intelligence can be improved, found that students labeled retarded lack the ability to orient themselves to other objects, identify problems, see relationships, make comparisons, and/or plan and organize; Logo can be used to help develop those abilities. Teachers must work with students, help students to define problems in simple terms, require work records, and encourage students to develop their own projects and long range goals. Logo is built upon a compelling educational philosophy, and offers a way to achieve difficult but worthwhile educational goals. Five references are listed. (LMM)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED254199

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☒ This document has been reproduced as received from the person or organization originating it.
☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

Logo's Problem-solving Potential

Evelyn J. Dale

820 Christie Court
Davis, California 95616

(916) 753-4413

IR011511

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Evelyn J. Dale

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

Introduction

Today, everybody wants computers in their school. They're the latest status symbol, an indicator that you are au current in the educational world. Getting the technology, in spite of the economy, is becoming less of an obstacle for many schools. With the arrival of computers it becomes clear that the real problem is not how to get it, but what to do with it.

The question that needs to be answered is, "What educational end do you hope to achieve with computers? The answer provides a philosophical direction that affects everything else that is done with the technology, including teacher training, software selection and, of course, student learning.

Many maintain that computers are essential to ensure graduates a place in the job market. Others point to its value as a remediation device. Neither response is wrong. They are just not enough. So, how do we find the answer to this question of computer purpose?

Let's start with one thing that seems pretty certain. The world ahead is changing and the future is difficult to predict. Assuming this to be true, consider the following questions. Will computer languages we know and teach today still be around much less used in five to ten years? How important will it be to know BASIC or any other computer language? Which will be more important to learn, how to find information or specific facts and figures? Do we want to teach job-related or

BEST COPY

broad-based knowledge? Should our students leave school as specialists or generalists? Answers to these questions provide the answer to the larger question of educational purpose. I would like to offer one response, and suggest how computers can be used to achieve the educational goal the response demands.

Given the uncertainty of the future and rapidity with which computer technology is changing, I must cast my vote in favor of a generalist position which insists that learning how to think and solve problems is the foundation upon which education should rest, and that the ultimate goal of learning is the desire to learn more. This should not be construed as a rejection of the basics. It is only saying that basic learning needs to be an integral part of the broader educational purpose, a means to an end, not the educational end itself.

Levels of Thinking

Let me digress a moment to discuss a crucial issue, and that is thinking. In his Taxonomy of Educational Objectives (1956), Benjamin Bloom describes six major categories of thinking: (1) knowledge, (2) comprehension, (3) application, (4) analysis, (5) synthesis, and (6) evaluation. Knowledge and comprehension, as we can see, are at the lower end of this scale. They represent the early stages of thinking, while synthesis and evaluation represent thinking on its most sophisticated level.

Most teachers acknowledge that education should help students to think on all six levels, but in reality that is not what happens. Research indicates that elementary school children

BEST COPY

spend 90% - 95% of their time learning at the knowledge level. High school students fare even worse, for they spend 95% - 100% of their time at this most basic level of learning.

What is Logo?

It is the teacher who has the greatest responsibility for seeing that computers promote higher levels of thinking and intellectual discovery. How can this be done? Where is software that comfortably meets these requirements? The beginning of an answer, I believe, is Logo. Logo is a computer language that reflects the learning theories and research of Jean Piaget, Seymour Papert, and artificial intelligence. It promotes learning by being accessible, interactive and extensible.

Logo's ease of access comes from the connection it makes between its beginning commands of FORWARD, BACK, RIGHT, and LEFT and the familiar body movements of forward, back and turning. These and other Logo "primitives" move a "turtle" about on a video screen or the floor the way people move about in space. With such familiar commands it is easy to control the computer almost immediately.

Logo is also interactive. That means Logo is responsive. It can either respond immediately when a line is typed in, or later, when a defined program is run. Interactive computer languages with descriptive error messages are not only user-friendly, but give structure and clarity to the programming and problem solving process.

Finally, Logo's extensibility gives the language versatility. Primitives can be used to define ideas and theories

BEST COPY

which can then be tested. On a very simple level FORWARD and RIGHT can define different geometric shapes such as a square, triangle or circle. On a more advanced level, they can be used to demonstrate the trajectory of a ball thrown at a particular velocity on a planet with a particular gravitational pull. And on a still more advanced level, they can help simulate the Theory of General Relativity (Abelson, and di Sessa, 1980).* It is this extensibility that is crucial for those who wish to challenge students of all abilities to think.

Logo Uses All Levels of Thinking

For example, consider how even elementary school children use Bloom's six levels of thinking when working with Logo. It is obvious that without knowledge nothing else can get started. Logo's basic knowledge is that of body movement, something most children have experienced from their very earliest years. For this reason children have little difficulty applying this knowledge within minutes of sitting down at the computer. Notice that I have said "applying." That means children are at Bloom's third level of thinking after only a few minutes with Logo. Using knowledge, of course, includes Bloom's first two levels of cognitive development, and so Logo provides an environment wherein children are almost immediately working with three out of Bloom's six levels of thinking. Already this is one step better than we manage with the usual classroom tasks. There

---

* Professor James Hurley of the University of California, Davis and myself used Logo to teach physics to a group of fifth and sixth grade children during January and February 1984.

BEST COPY

aren't too many educational materials on the market that let that happen.

What about the other, more advanced categories of analysis, synthesis, and evaluation? Analysis, the next step up the scale, is defined by Bloom as "...the breakdown of the material into its constituent parts and detection of the relationships of the parts and of the way they are organized." (Bloom, 1956, p. 144). Procedures are the Logo structures that permit problems to be divided into modules. Of course, Logo commands can be written sequentially, but many children quickly learn that long, sequential programs without line numbers are very hard to fix, i.e., debug. Besides, using these procedural modules saves time. You can often use a procedure from one program as a subprocedure or superprocedure in another program. Kids soon learn that sharing procedures is not only time saving, its fun! Seeing direct and practical reasons for analyzing a problem lets children experience the value of this kind of thinking.

It is a fairly natural, and not too difficult, step to apply the analytical knowledge of a program's structure to the creation of a new program. This reorganizing of already existing procedures and, perhaps, writing of new ones calls for a more complex level of thinking, what Bloom calls synthesis.

Bloom's sixth and final category is evaluation. Here value judgment depends upon clearly defined criteria and the student's understanding of all the other categories. It is, of course, quite common for students to evaluate their classmates' projects but such judgments are often vague and subjective. Logo projects

BEST COPY

involve a certain amount of evaluation from the outset. More important, however, is ongoing evaluation through the debugging process. Debugging involves operation of a program, criticism of its operation, finding and fixing errors, and revision to improve operation. This means that evaluation is an integral part of the whole process of developing Logo projects. It is also an important part of concluding one project and beginning another.

What's Needed to Build Thinking Skills

The big question is how to develop higher levels of thinking. Reuven Feuerstein, an Israeli psychologist and Professor at Bar Ilan University, has been concerned with diagnosing and improving thinking skills since the end of World War II. In 1950 he became involved in the screening of Jewish children as they entered the new state of Israel. These children were given a series of standardized intelligence tests in an effort to assess their intellectual abilities. Feuerstein found that many of these young people tested well below average in intelligence. Surely they all weren't slow learners! He began to wonder if their scores weren't a result of childhood deprivation. The standardized tests were unable to answer this question because they tested only what children knew not their learning potential (Hobbs, 1980).

Since that time, Feuerstein and his colleagues have developed the Learning Potential Assessment Device (LPAD) to evaluate learning potential and an Instrumental Enrichment (IE) program to develop specific thinking skills. The importance of

BEST COPY

Feuerstein's work, however, goes beyond these diagnostic and prescriptive tools because it is based on the important assumption that intelligence, rather than being unvariable, is malleable and can be improved.

What Feuerstein found, as he worked with these "retarded performers," is that they lack specific skills necessary to solve problems but that they can, in many cases, acquire them. For example, he found retarded performers lack the ability (1) to orient themselves to other objects, (2) identify problems, (3) see relationships, (4) make comparisons, and/or (5) plan and organize (Chance 1981).

I would like to suggest that Logo can be used to help develop all five of these abilities. For example, work with spatial orientation begins as soon as the turtle is moved about the screen. This thinking skill continues to be developed as shapes are created and placed on the screen. Debugging Logo programs provides plenty of practice identifying problems. Moreover, Logo's interactive and procedural structure makes the process systematic and thus easier to understand. Using proceduralization to write and follow programs also helps to develop the ability to see relationsips. For instance, each Logo program provides a special context in which separate parts interact to produce an integrated whole. In another context, the same parts can produce quite different results. Logo also invites careful examination of shapes, language, and programs which helps to develop the vocabulary and skill to make comparisons. Finally, Logo's procedural structure can develop

BEST COPY

planning and organizational skills that use inductive and deductive thinking. It is important to understand that Logo supports both of these methods of thinking. Many people consider deductive or top down thinking to be best. Such a perspective denies the importance of inductive thinking and limits the learner's repetoire of problem solving tools. If fact, solutions to problems of any complexity usually combine both styles of thinking.

Self-esteem is important too. A positive self-image improves learning. Feuerstein found retarded performers tend to be passive with low self-esteem. They do not trust their intellectual abilities and prefer to rely on others (Chance, 1981). Logo improves the learner's self-image because it invites active participation and independence. This was demonstrated in a Stanford University study conducted by Dr. James Milojkovic. Fifth and sixth grade students programming with Logo were found to have more self-confidence and independence than those using BASIC or no computer language at all. This sense of self-worth also spilled over into the regular classroom where the Logo programmers showed a more positive attitude toward school and learning in general than the other two groups (Milojkovic, 1983).

The Teacher's Role

Well, that seems pretty easy. All teachers need to do is bring their students and Logo together, and sit back and watch the learning take place! Not quite. With Logo, teachers are essential and cannot take a back seat to Logo or any other piece

BEST COPY

of technology. The kind of teaching, however, changes. Instead of directing, there is guiding and suggesting. Instead of knowing everything there is stretching of intellectual and imaginative limbs. Knowing all the answers isn't enough, a willingness to seek out the difficult and unknown is required. In short, the teacher becomes a model learner who is willing to experience all the excitement, drudgery, and effort that accompanies indepth learning.

It is said that the Logo teacher uses non-traditional teaching methods and so it is assumed that these methods are somehow new. They aren't. It's just that many teachers, caught up in the everyday demands of time schedules and specific curricular, cannot always apply the good teaching strategies they know. Here are some specific teaching strategies that help develop problem solving abilities and work well in a Logo environment. (1) Work with students. Help them to transform and reorder information. (2) Help students to define problems in simple terms. (3) Require work records such as Logo journals. These are not for grades but to help guide cognitive development. (4) Encourage students to develop their own projects and long range goals. (5) Whenever possible, answer questions with questions that guide and stimulate thinking.

Teachers also need to help students develop their own strategies for solving problems. Here are a few that provide a useful beginning. (1) Guess and check. (2) Act out the problem. (3) Draw a picture. (4) Look for patterns. (5) Work backwards. (6) Solve separate parts of the problem. (7) Apply logical

BEST COPY

reasoning. (8) Make an organized list. (9) Make tables. (10) Use tree diagrams.

With Logo programming projects, students are able to test the value of each of these strategies. Consider the simple example of drawing a square. Learners may not know how to teach the turtle to draw a square but by guessing and checking they may discover the answer. If they can't solve the problem, they can act it out by Playing Turtle or drawing a picture. Once they teach the turtle to draw the square, they can look for the FORWARD/RIGHT pattern in the code. If anyone is still having trouble drawing a square, perhaps they should change tactics and try working backwards, or working with the separate parts (distance and angle) of the square. Applying logical reasoning helps, especially when it takes the form of making an organized list or table of all the different angles used to try to draw the square. The solution to this problem certainly doesn't require a tree diagram, but it can always be used to clarify the process and indicate new design projects based on the square.

There is one more important role for teachers. Logo is more than a computer language. It is a means to the larger educational goal of learning how to think and solve problems in a variety of contexts. In order to reach this goal, students need to be able to apply what they learn with Logo to different, non-computing contexts. This means bridging experiences need to be provided to help students transfer learning from one context to another.

Learning does not transfer easily so teachers should select

BEST COPY

experiences carefully and then lead structured discussions on the relationship between these different experiences. How did you organize your Logo program? Can you use some of your programming ideas to organize cleanup? Who can divide our task into major categories? How are our programs for Logo and cleanup alike? Is there a better way to organize the tasks? These are all questions that help students transfer learning.

Developing higher levels of thinking also takes time. Feuerstein finds it takes one to two years before changes in thinking patterns begin to emerge (Chance, 1981). For this reason, teachers need to be patient, recognizing that significant results are not likely to materialize immediately.

Conclusion

I am not outlining an easy way to effective teaching. Logo is only a computer language, but it is built upon a compelling educational philosophy. What Logo does offer is an exciting means with which to achieve difficult but worthwhile educational goals.

Bibliography

Abelson, Harold and Andrea di Sessa. Turtle Geometry: The Computer as a Medium for Exploring Mathematics, Cambridge, Ma.: The MIT Press, 1980.

Bloom, Benjamin S., ed. Taxonomy of Educational Objectives, New York, Longmans, Green and Co., 1956.

Chance, Paul. "The Remedial Thinker," Psychology Today, Oct. 1981.

Hobbs, Nicholas. "Feuerstein's Instrumental Enrichment; Teaching Intelligence to Adolescents." Educational Leadership, April 1980.

Milojkovic, James. "Children Learning Computer Programming: Cognitive and Motivational Consequences." Doctoral dissertation, Stanford University, 1983.

BEST COPY